



Distance Learning System

MySql programiranje i administracija

Uvod u SQL

Šta je SQL

- SQL je jedini jezik baza podataka
- SQL je u širokoj primeni. Još od kako je predstavljen, ranih 70-ih godina prošlog veka, bio je kritikovan, menjan, proširivan, ali na kraju i prihvaćen od strane svih relevantnih činilaca na tržištu baza podataka.
- SQL je postao standard Američkog Nacionalnog Instituta za standarde (ANSI) 1986. godine i Internacionalne Organizacije za standarde (ISO) 1987. godine.
- Uprkos standardizaciji, SQL kod različitih sistema za upravljanje bazama podataka nije u potpunosti kompatibilan, pošto svaki proizvođač implemetira SQL sa određenim odstupanjima i primenom specifičnih proširenja.
- SQL se primarno koristi za formulisanje instrukcija sistemu baze podataka, koje obuhvataju naredbe za manipulaciju bazama i podacima.

SQL

SQL Sintaksa

- SQL nije osetljiv na velika i mala slova;
- Svaki SQL upit (ukoliko ih ima više) mora da se završi oznakom ; (ovaj separator se može promeniti);
- Stringovi se pišu isključivo u navodnicima; MySQL prihvata i jednostruke i dvostruke navodnike za string, ali je najbolja praksa koristiti jednostruke;
- Objekti u SQL-u se mogu staviti u Backtick navodnike (select * from `my table`);
- promenljive u SQL-u uvek počinju oznakom @.

Tekstualne konstante

- String predstavlja sekvencu bajtova ili karaktera, koji su ograničeni ili jednostrukim navodnicima (') ili dvostrukim navodnicima (").
- Generalno, postoje dva tipa stringova u MySQL-u i to su **binarni stringovi** i **nebinarni stringovi**, odnosno stringovi koji su sastavljeni od karaktera. Razlika između ove dve vrste stringova je u osnovnoj gradivnoj jedinici koja je za binarne stringove, naravno, bajt, dok je za nebinarne stringove to karakter. Još jedna razlika je veoma bitna, a to je činjenica da se za binarne stringove ne može definisati set karaktera ili kolacija, s obzirom da se čuvaju kao sekvenca bajtova

Komentari

- MySQL server podržava tri vrste komentara:
- jednolinijski komentar, navođenjem karaktera **#**
- jednolinijski komentar navođenjem karaktera **–**
- višelinijski komentar koji je kreiran na isti način kao i u programskom jeziku C, korišćenjem karaktera **/* i */** gdje sve između ovih oznaka predstavlja komentar
- MySQL server poznaje i jednu vrstu specijalnih komentara, koji se nazivaju uslovni komentari. **/*! MySQL-specific code */**

SQL Naredbe

- SQL naredbe se dele na tri seta
 - Data Definition Language (**DDL**)
 - CREATE, ALTER DROP
 - Data Manipulation Language (**DML**)
 - SELECT, INSERT, UPDATE i DELETE
 - Data Control Language (**DCL**)
 - GRANT i REVOKE

Naredbe za definiciju

- Definicione SQL naredbe (DDL) čine tri naredbe: **CREATE**, **ALTER** i **DROP**. Isključivo se tiču strukture, a ne samih podataka, što znači da, uz pomoć njih nikada nećete dodavati ni jedan jedini podatak u bazu.
- Kada želite da kroz SQL skriptu stvorite bazu podataka, ili neke njene entitete, koristićete upravo ovaj set naredbi

CREATE

- Svaki objekat koji se pravi u bazi, pravi se ovom naredbom, počevši od same baze, preko korisnika, pa do tabela, ključeva, indeksa...
- Sintaksa naredbe je:
CREATE object_type object_name
- Primer:
 - ***CREATE DATABASE my_database;***
 - ***CREATE TABLE mytable (id int primary key auto_increment, name varchar(40))***

Vežba

- Potrebno je kreirati tabelu (**cars**), koja će imati kolone: **car_id, car_make, car_model, car_price**
 - Kolona car_id mora biti primarni ključ, i imati auto inkrement
 - kolone **car_make i car_model** moraju biti tekstualnog tipa
 - Kolona car_price mora biti decimal sa mogućnošću unosa šestocifrenog broja sa dve decimale
-
- sačuvati kreiranu tabelu za sledeći primer

ALTER

- Naredba ALTER služi da neki, već napravljeni, sadržaj zamenimo. Kada kažemo napravljeni, podrazumevamo da je napravljen naredbom CREATE
- ALTER naredba se obično koristi u jednom od dva oblika. U jednom, ona se ponaša kao naredba CREATE i prihvata kompletnu novu definiciju objekta, sa izmenama. U drugom, potrebno je eksplicitno naglasiti izmene i samo njih uneti kroz odgovarajuće modifikatore. Sami oblici zavise od tipa objekta nad kojim se alter izvršava.

CREATE VIEW my_view AS select 'my_view'

ALTER VIEW my_view as select 'my_edited_view'

CREATE TABLE my_table (id int, name varchar(50))

ALTER TABLE my_table ADD COLUMN surname varchar(50)

Vežba

- Tabeli kreiranoj u prethodnom primeru, treba dodati kolonu stock, koja predstavlja broj automobila na stanju
-

DROP

- DROP se koristi za brisanje objekta i kao parametre prihvata tip i naziv objekta:

DROP TABLE my_table

DROP DATABASE my_first_db

DROP INDEX my_idx

- Često se koristi u kombinaciji sa IF EXISTS:

DROP DATABASE IF EXISTS my_database;

CREATE DATABASE my_database;

Vežba

- Obrisati tabelu **cars** iz prethodnog primera

Vežba

- Potrebno je kreirati tabelu users koja će pamtit i sledeće podatke: id korisnika koji će se povećavati automatski za jedan prilikom svakog unosa i koji će biti primarni ključ tabele, ime i prezime korisnika.
- Potrebno je kreirati i dodatni upit koji kreiranoj tabeli dodaje još jedno polje sa šifrom korisnika.

Rešenje

<https://bitbucket.org/snippets/vmaric/6n87GL>

create table users

(

user_id int primary key auto_increment,

user_firstName varchar(50),

user_lastName varchar(50)

)

alter table users add column password varchar(15)

Vežba

- Potrebno je kreirati skriptu za kreiranje baze podataka application_06. Ova skripta će brisati postojeću bazu ukoliko postoji. Takođe, potrebno je kreirati skriptu za kreiranje tabele computer_parts, tako da obriše postojeću istoimenu tabelu ukoliko postoji. Ova tabela treba da sadrži podatke o ID-u uređaja, šifri uređaja, tipu uređaja, nazivu uređaja i ceni uređaja.

Rešenje

```
drop database if exists application_06;  
create database application_06;  
drop table if exists computer_parts;  
create table computer_parts  
(  
    computer_part_id int primary key auto_increment,  
    computer_part_code varchar(5),  
    computer_part_type int,  
    computer_part_name varchar(256),  
    computer_part_price decimal(7,2)  
)
```